

DUBLIN INSTITUTE OF TECHNOLOGY
KEVIN STREET, DUBLIN 8.

BSc Information Technology

Year 3

SUPPLEMENTAL EXAMINATIONS 2006

DISTRIBUTED INFORMATION SYSTEMS

MR. C. O'LEARY
DR. B. O'SHEA
MR. P. LAWLESS

TIME ALLOWED: 3 HOURS

ATTEMPT **4** QUESTIONS

ALL QUESTIONS CARRY EQUAL MARKS

1. (a) In relation to XML and distributed applications, comment upon the importance, as you see it, of *well-formedness* and *validity* (through DTD [Document Type Definition] and XML Schema). **(6 marks)**
- (b) The *semantic web* makes extensive use of XML technologies.
- Present an informed argument for or against the potential of the semantic web to revolutionise distributed computing. **(9 marks)**
- (c) Describe clearly the role of each of the following in the web service infrastructure:
- HTTP
 - XML
 - SOAP
 - WSDL
 - UDDI
 - XML Security
- (10 marks)**
2. (a) Distinguish clearly between *external synchronisation* and *internal synchronisation* of computer clocks, and comment on the types of applications that should make use of each type of synchronisation. **(6 marks)**
- (b) Describe *Cristian's method* and the *Berkeley method* for clock synchronisation. Comment on the relative merits of each method. **(9 marks)**
- (c) Using a clear example, demonstrate the advantage of *Vector clocks* over standard *Lamport clocks*. **(10 marks)**
3. (a) Define *serial equivalence* in terms of concurrent transactions. **(6 marks)**
- (b) Demonstrate how *two-phase locking* can be used to ensure serial equivalence. Show also how this can be extended to prevent problems such as *dirty reads* and *premature writes*. **(9 marks)**
- (c) Describe in detail, using examples, how *nested transactions* differ from *flat transactions*.
- What impact do these differences have on the locking mechanisms required for these transactions? **(10 marks)**

4. (a) Using an example, demonstrate the operation of the Router Information Protocol (RIP). **(6 marks)**
- (b) In relation to IP Addressing, describe the *motivation for* and the *operation of* each of:
- Classless Inter-Domain Routing (CIDR)
 - Network Address Translation (NAT)
- (9 marks)**
- (c) Modern wireless, mobile and ubiquitous computing environments require a rethink of some of the fundamental approaches to routing and communications in what are called *Personal Area Networks*.
- Within this context, illustrate the operation of the IEEE 802.15.1 standard (*Bluetooth*). **(10 marks)**
5. (a) What, in your opinion, is the fundamental issue that makes the design and implementation of distributed computing applications more complicated than non-distributed applications?
- Justify your opinion. **(6 marks)**
- (b) Compare and contrast *distributed object systems* and *non-distributed object systems* under the following headings:
- Object instantiation
 - Method invocation
 - Parameter passing
 - Memory allocation / deallocation
 - Failure
- (9 marks)**
- (c) Provide a design for a generic distributed event based framework using object oriented concepts. **(10 marks)**

6. (a) Examine the code below. What output would you expect when this code is compiled and run? Why?

```
public class Q6a {

    public Q6a() {
        Printable book1 = new Book("History of Eire", "John Starr", "0-00-567453-8", 3);
        Printable book2 = new Publication("Da Vinci Code", "Dan Brown");
        Book book3 = new Book("Mapping the Earth", "Thar Saile", "0-00-432434-7", 2);
        book3.addChapterTitle(1, "Europe");
        book3.addChapterTitle(2, "Africa");

        book1.print();
        book2.print();
        book3.print();
    }

    public static void main(String args[]) {
        new Q6a();
    }
}

interface Printable {
    public void print();
}

class Publication implements Printable {
    protected String title;
    protected String author;

    public Publication(String title, String author) {
        this.title = title;
        this.author = author;
    }

    public void print() {
        System.out.println("Publication: " + title + ", by " + author);
    }
}

class Book extends Publication {
    private String ISBN;
    private int numberOfChapters;
    private String chapterTitles[];

    public Book(String title, String author, String ISBN, int numberOfChapters) {
        super(title, author);
        this.ISBN = ISBN;
        this.numberOfChapters = numberOfChapters;
        this.chapterTitles = new String[numberOfChapters];
    }

    public void addChapterTitle(int chapterNumber, String title) {
        chapterTitles[chapterNumber - 1] = title;
    }

    public void print() {
        System.out.println("Book: " + title + " (ISBN: " + ISBN + "), by " + author);
        for(int i = 0; i < numberOfChapters; i++)
            if(chapterTitles[i] != null)
                System.out.println("\tChapter " + (i + 1) + ": " + chapterTitles[i]);
    }
}
```

(6 marks)

- (b) Using diagrams and sample code, show precisely how you would design and implement (in Java) a chat application with the following characteristics:

Clients connect to a centralised server. Clients can start a new chat room by supplying the theme (topic) for the room. Other clients can then lookup rooms by their theme. Once clients have either created a room or looked up a room, they can then send a message into the room. Clients can also ask for the contents of the last message sent to the room.

You may use pseudocode for the detail, but you must clearly identify the required classes and interfaces.

(Note that clients must poll the server to receive the last message.)

(9 marks)

- (c) Expand your solution from part (b) so that clients are informed by the server when a new message is sent into the room.

You may use pseudocode for the detail, but you must clearly identify the required classes and interfaces.

(Note that this removes the requirement for polling.)

(10 marks)